

VISUALISASI TIGA DIMENSI (3D) REAL TIME MENGUNAKAN OPENGL

Naskah Publikasi



diajukan oleh

TOHIR ISMAIL
00.21.0048

kepada

**SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
STMIK AMIKOM YOGYAKARTA
2010**

NASKAH PUBLIKASI

**VISUALISASI TIGA DIMENSI (3D) REAL TIME
MENGUNAKAN OPENGL**

disusun oleh

**TOHIR ISMAIL
00.21.0048**

Dosen Pembimbing



**Abas Ali Pangera, Ir. M.Kom
NIK. 190302008**

Tanggal 3 Juni 2010

**Ketua Jurusan
Teknik Informatika**



**Abas Ali Pangera, Ir. M.Kom
NIK. 190302008**

REAL TIME THREE DIMENSION (3D) VISUALIZATION USING OPENGL

VISUALISASI TIGA DIMENSI (3D) REAL TIME MENGUNAKAN OPENGL

Tohir Ismail

Jurusan Teknik Informatika

STMIK AMIKOM YOGYAKARTA

ABSTRACT

Three dimensional (3D) visualization technology has been growing rapidly, from the modelling technology, animation, texturing, special effects, rendering, and the divergence usage of this technology. 3D visualization is not used in animation feature film and architectural design only anymore, instead, already adopted in many industries such as medical institution, education, military, laboratories, automotive, transportation, broadcast and many others.

Generally, the process to creating a 3D visualization is by modelling 3D objects and their environments in a 3D authoring software package or CAD (Computer Aided Design) software package, modifying objects parameters, adding cameras, lights, applying materials, special effects and then enter into rendering process that consumes much time before finally a 3D visualization output ready for viewing.

In this thesis, researcher try to develop a software that could do a real time 3D visualization, no need for waiting, a parameter changes to the 3D objects directly reflected in 3D visualization display. The objects parameters changes can be driven by a local data file or remote controlled via Client-Server data communication. With the help of OpenGL library, this software is ready for portability in case needed to run on another platform. Hopefully with use of this software could improve the speed and ease in decision making process as improved experience by reviewing real time data representation in 3D visualization.

Keywords : Three dimensional (3D) visualization, Real Time, Rendering, OpenGL

1. PENDAHULUAN

Teknologi komunikasi manusia mengalami perkembangan yang sangat cepat dalam beberapa dekade terakhir. Komunikasi yang semula sangat terbatas oleh parameter jarak dan waktu, dimana manusia hanya bisa berkomunikasi dengan manusia lain ketika berada dalam jarak tertentu dan waktu yang sama, kini tidak ada lagi parameter jarak dan waktu, manusia tetap bisa berkomunikasi meski dengan jarak yang sangat jauh dan dalam waktu yang berbeda. Cara manusia menyampaikan pesan komunikasi jarak jauh pun mengalami perkembangan, dari hanya sekedar pesan teks singkat, berkembang menjadi pesan audio visual lengkap dengan pengembangan citra visualnya, mulai dari video nyata dari orang yang berkomunikasi tersebut sampai ke karakter virtual baik dua dimensi (2D) maupun tiga dimensi (3D) sebagai representasi dari para komunikatornya.

Komunikasi audio visual telah berkembang fungsinya dari hanya sebatas alat komunikasi sosial sehari-hari antar manusia menjadi sebuah industri bernilai milyaran rupiah. Pemanfaatan teknologi komunikasi audio visual ini telah menjadi standar dalam industri penyampaian pesan dari perusahaan untuk memperoleh pelanggan sebanyak-banyaknya. Demikian juga dalam industri hiburan, audio visual masih menjadi pilihan favorit baik bagi penyedia hiburan maupun bagi masyarakat yang menjadi sasaran industri hiburan tersebut. Dalam dunia pendidikan, audio visual juga menjadi sarana yang sangat ampuh dalam menyampaikan sebuah ilmu pengetahuan, ketika suatu pengetahuan kurang bisa dipahami jika hanya disampaikan melalui tulisan, maka dengan media audio visual menjadi lebih mudah dipelajari.

Mengingat masih efektifnya cara berkomunikasi melalui audio visual ini, membuat semakin banyak orang berlomba-lomba menuangkan ide dan kreatifitas mereka dalam bentuk audio dan visual dengan cara-cara yang sebelumnya belum pernah dikerjakan oleh orang lain. Riset terus dikembangkan guna menemukan metode-metode baru dalam memvisualisasikan ide kedalam sebuah karya audio visual.

Sistem komunikasi visual telah diadopsi hampir di semua bidang industri, dan telah mengalami pengembangan dari hanya sebatas visualisasi grafik dua dimensi, menjadi visualisasi tiga dimensi yang mampu merepresentasikan secara lebih detail mengenai sesuatu. Teknologi visualisasi tiga dimensi yang awalnya hanya digunakan untuk mendesain produk dengan bantuan komputer (CAD, *Computer Aided Design*) dan untuk industri permainan atau games, kini telah menjadi standar di berbagai bidang untuk membantu dalam pengambilan keputusan atau alat bantu peraga *virtual*, contoh dalam industri property, visualisasi tiga dimensi digunakan sebagai peraga *virtual* pengganti atau komplemen dari sebuah maket, dalam industri militer, visualisasi radar tiga dimensi

mampu mempercepat pengambilan keputusan, dalam industri pendidikan, visualisasi tiga dimensi dijadikan sebagai alat simulasi berbagai ilmu pengetahuan, dan sebagainya.

Riset-riset mengenai cara mempermudah memvisualisasikan ide atau data secara lebih cepat dan akurat telah banyak dilakukan, khususnya teknologi visualisasi tiga dimensi (3D) telah mengalami perkembangan yang sangat pesat, misalnya dengan diciptakan teknologi motion capture, menggerakkan obyek 3D menjadi lebih mudah, facial capture membuat animasi ekspresi wajah menjadi lebih mudah dan akurat, scanner 3D membuat proses pemodelan obyek tiga dimensi (3D) menjadi lebih cepat dan akurat, software-software *authoring* tiga dimensi yang lengkap dengan obyek-obyek 3D siap pakai (*pre-made*) serta sistem template 3D mempercepat desain suatu model obyek tiga dimensi.

Namun pada umumnya software-software *authoring* tiga dimensi tersebut tidak mendukung representasi visual tiga dimensi (3D) dari data eksternal secara real time, melainkan harus melalui proses *rendering* untuk menjadi sebuah file video yang baru bisa diputar ketika proses *rendering* selesai. Ketika perubahan parameter obyek dilakukan terhadap sebuah visualisasi 3D, maka perlu proses render yang cukup memakan waktu sebelum hasil akhirnya bisa dilihat.

Perlu dikembangkan sebuah software yang mampu merepresentasikan visual dalam format tiga dimensi secara real time dengan memasukkan data atau parameter-parameter yang diperoleh dari suatu alat eksternal, tanpa perlu proses rendering ke sebuah file video, melainkan langsung bisa di-*display*-kan di layar.

2. LANDASAN TEORI

2.1. TEORI DASAR

Dalam pengembangan software visualisasi tiga dimensi secara real time menggunakan OpenGL, disini digunakan beberapa istilah atau terminologi dan teknologi standar, diantaranya yaitu.

2.1.1. Real Time (Computer Graphics)

Tujuan utama komputer grafis adalah membangkitkan gambar hasil komputer (*computer generated image*) ke layar dengan aturan tertentu. Gambar ini sering disebut juga *Frame*, seberapa cepat gambar bisa dihasilkan dan ditampilkan di layar menentukan tingkat real time aplikasi tersebut. Secara umum real time dalam bidang komputer grafis adalah untuk merujuk ke sistem yang tidak menggunakan metode tradisional *offline*

rendering, sistem tradisional ini memerlukan waktu beberapa detik, menit bahkan jam atau hari untuk bisa menghasilkan satu *frame* gambar, sebagai akibat digunakannya metode-metode rendering ekstensif dan belum mampu dikerjakan secara cepat oleh *hardware* atau *resource* komputer saat ini. Real time dalam komputer grafis umumnya digunakan untuk menyebut sistem yang mampu menampilkan satu frame dalam waktu 1/30 detik atau lebih cepat, dengan memanfaatkan GPU (*graphics processing unit*), selain prosesor utama dalam komputer.

2.1.2. OpenGL

OpenGL pada awalnya didesain untuk kepentingan CAD (*Computer Aided Design*) dan 3D API (*Three Dimension Application Programming Interface*) yang dibuat oleh Silicon Graphics, Inc. pada tahun 1992 untuk platform UNIX menggunakan X-Terminal. OpenGL semula digunakan khusus secara tertutup untuk sistem proprietary mereka, untuk kepentingan industrial, mekanikal dan analisis scientific.

Namun dalam perkembangannya OpenGL API banyak digunakan oleh para pembuat games seiring dengan dirilisnya sistem operasi Windows oleh Microsoft pada awal tahun 1996, ketika industri hardware 3D accelerator mulai masuk, SGI merevisi aturan lisensi penggunaan 3D API mereka sehingga lebih terbuka, menguatkan posisinya sebagai *cross-platform* API untuk membangun aplikasi grafis interaktif 2D dan 3D paling dominan.

OpenGL adalah merek dagang OpenGL Architectural Review Board (ARB) yang didirikan pada tahun 1992 sebagai pengelola dan pengembang OpenGL, kemudian pada September 2006 berubah menjadi OpenGL Working Group dibawah Khronos Group konsorsium untuk API standar terbuka.

OpenGL mendefinisikan status mesin dalam proses rendering, atribut-atribut status mesin diubah melalui pemanggilan prosedur. OpenGL API didesain untuk mengakomodasi teknik rendering grafis tingkat lanjut, seperti texture mapping (menempelkan tekstur/gambar ke permukaan grafis), anti-aliasing, transparansi, kabut, pencahayaan, motion blur, spesial efek, dan transformasi obyek 3D.

Dengan memanfaatkan OpenGL API, operasi matematis transformasi dan operasi vektor 3D menjadi sangat mudah dan cepat, dibanding dengan membangun kode sendiri dari awal. Programmer bisa lebih fokus dengan content dibanding dengan low level hardware programmingnya.

Mengingat penerimaan industri yang sangat luas terhadap OpenGL sehingga Microsoft juga memasukkan OpenGL API sebagai bagian dari Operating System mereka atas lisensi dari SGI, meskipun sebenarnya Microsoft memiliki teknologi grafis 2D dan 3D

tersendiri yang bersaing dengan OpenGL, yaitu Direct3D dan DirectX. Meski demikian SGI tetap mengembangkan versi tersendiri dari OpenGL yang khusus untuk versi Windows diberi nama CosmoGL (sekarang berubah menjadi hanya OpenGL). OpenGL API dari Microsoft dan dari SGI saling compatible, hanya masalah optimasi pada tingkat hardware yang berbeda implementasinya.

Pada level hardware (kartu grafis), OpenGL API diwujudkan dalam bentuk MCD (*Mini Client Drivers*) dan ICD (*Installable Client Drivers*), sehingga tergantung pada vendor kartu grafis untuk memanfaatkan API yang mana, namun secara teknologi ICD lebih bagus performanya dibanding MCD.

OpenGL didukung pada hampir semua sistem operasi seperti Windows9x, NT, 2000, XP, Vista, MacOS, OS/2, BeOS, Linux, berbagai varian UNIX dan Mobile Platform.

2.1.3. GLScene

GLScene adalah sebuah library visualisasi tiga dimensi (3D) berbasis OpenGL untuk bahasa pemrograman Pascal. GLScene bisa digunakan di lingkungan pengembangan (IDE, *Integrated Development Environment*) Embarcadero/Borland Delphi, FreePascal Lazarus maupun lingkungan pengembangan lain yang menggunakan bahasa pemrograman Pascal.

GLScene adalah wrapper dari library OpenGL untuk memudahkan programmer membangun visual tiga dimensi tanpa perlu mengetahui kode API (*Application Programming Interface*) dari OpenGL itu sendiri, semula dikembangkan oleh Mike Lischke (sampai dengan versi 0.5) kemudian pengembangan dan pemeliharaan diambil alih oleh Eric Grange sampai saat ini.

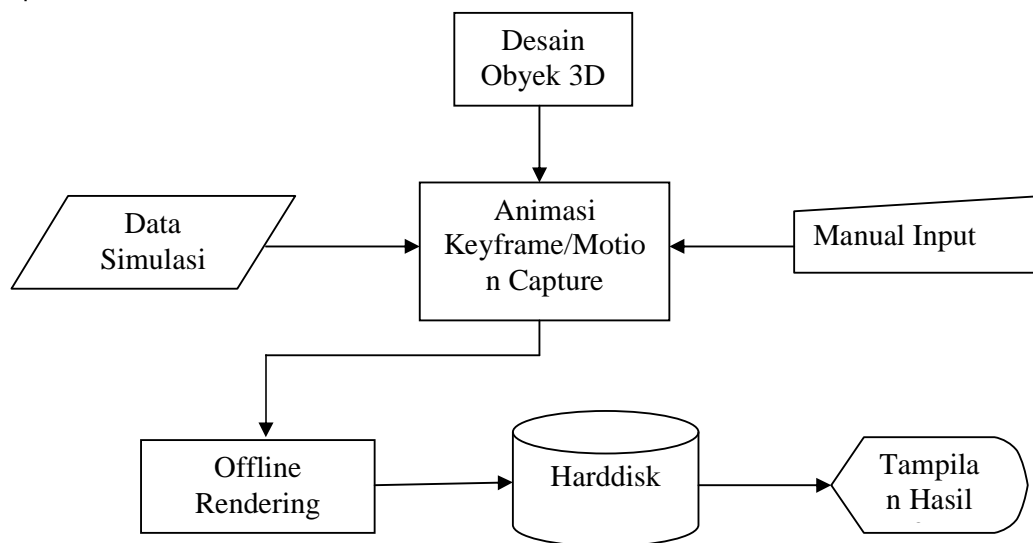
GLScene bersifat kode terbuka (*Open Source*) dan didistribusikan dibawah lisensi Mozilla Public License (MPL) yang kurang lebih bisa disimpulkan sebagai berikut.

- 1) Bebas digunakan oleh siapa saja, baik untuk membangun aplikasi gratis maupun komersial, serta berhak mendistribusikan ulang paket library dalam GLScene.
- 2) Tetap menjadi software yang memiliki Hak Cipta (*Copyrighted Software*), harus meletakkan kredit atau link ke website pengembang GLScene dalam software yang dibuat jika menggunakan komponen GLScene.
- 3) Segala perubahan atau modifikasi terhadap library GLScene harus dijadikan milik umum.
- 4) Kode atau library tambahan yang digunakan dalam software yang menggunakan komponen GLScene berhak untuk tetap menjadi hak privat atau komersial.

GLScene berisi komponen visual dan non visual yang terintegrasi dengan IDE meliputi obyek-obyek tiga dimensi (3D), material, antarmuka hardware dan software, animasi, rendering, suara dan utility lainnya. Sehingga secara keseluruhan GLScene bisa dikatakan sebagai mesin 3D generik (*Generic 3D Engine*) yang bisa digunakan untuk membangun berbagai aplikasi 3D seperti games, tutorial, CAD, virtual stages, visualisasi data, dan sebagainya.

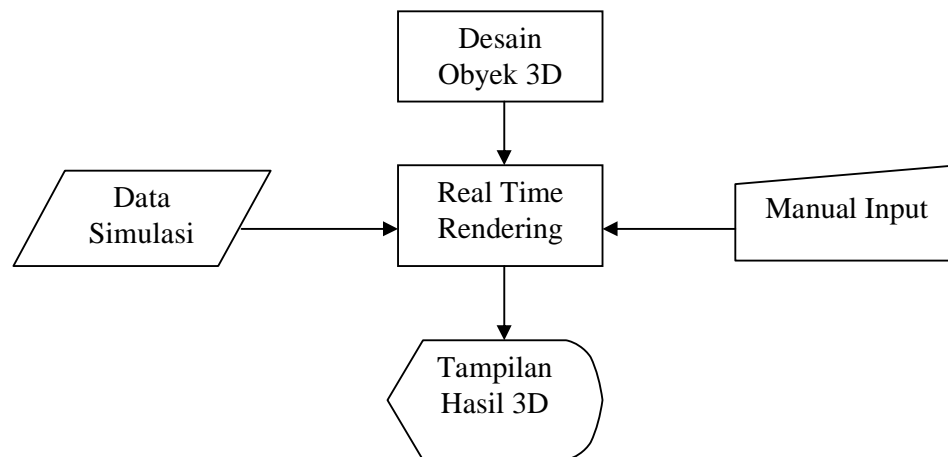
3. ANALISIS DAN PERANCANGAN SISTEM

Secara umum proses untuk menghasilkan visualisasi dan animasi tiga dimensi (3D) dari suatu kasus simulasi adalah dengan cara membangun model-model obyek 3D di sebuah software *3D authoring*, menganimasikan menggunakan metode *Keyframe* atau *Motion Capture* sesuai data atau parameter simulasi yang ada, me-render menjadi gambar atau video. Proses menganimasikan dan me-render dalam metode ini (gambar 3.1) memakan waktu cukup panjang bisa beberapa menit, jam atau bahkan hari. Hal ini tidak bisa diterima jika diperlukan pengambilan keputusan secara cepat saat data atau parameter sudah tersedia.



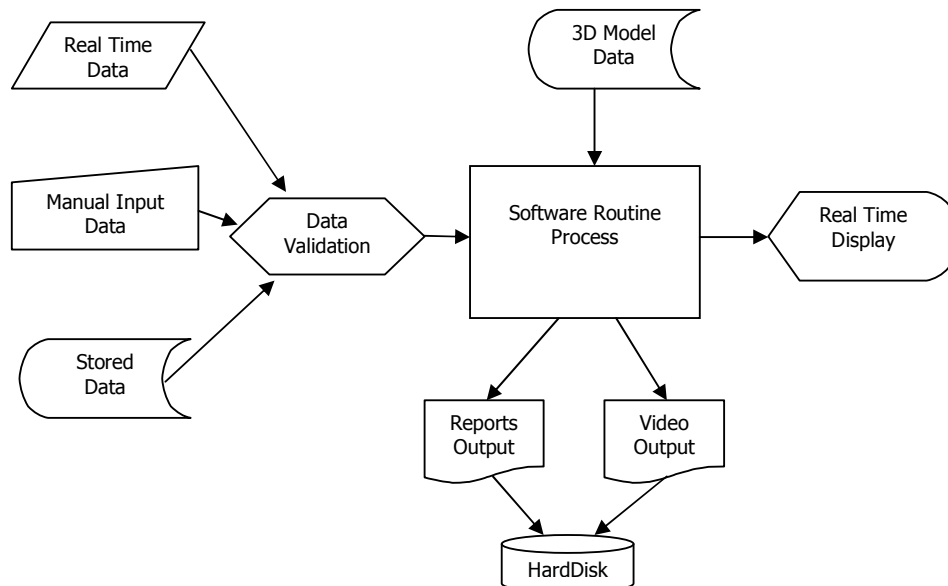
Gambar 3.1 Sistem visualisasi tidak real time

Usulan sistem yang akan dibuat adalah menghilangkan proses animasi manual serta proses rendering yang memakan waktu dengan memanfaatkan teknologi real time rendering (gambar 3.2).



Gambar 3.2. Sistem visualisasi real time

Sistem visualisasi tiga dimensi real time didesain sebagai sistem modular sehingga dengan mudah bisa ditambah atau dikurangi fitur dan obyeknya sesuai kebutuhan. Berikut flow chart sistem yang akan dikembangkan (gambar 3.3).



Gambar 3.3 Flow Chart Sistem

Dengan menggunakan metode analisis SWOT, bisa ditemukan banyak kekuatan (strength), kelemahan (weakness), peluang (opportunity) dan ancaman (threat), terdapat beberapa strategi yang dapat digunakan untuk mengatasi masalah-masalah tersebut.

a. Strategi SO

- Meningkatkan promosi pemasaran dan penetrasi pasar baru.
- Menemukan dan membangun nilai (*selling point*) produk yang paling kuat.
- Mengikuti tren perkembangan teknologi terkait.

b. Strategi WO

- Menemukan kelemahan-kelemahan produk pesaing dan menjadikan sebagai *selling point* produk.
- Memantau produk-produk pesaing yang bersifat kode terbuka (*open source*) dan memberikan dukungan (*support*) purna jual lebih baik dari pesaing.

c. Strategi ST

- Mencari partner pengembang hardware untuk membangun sistem terintegrasi.
- Menentukan strategi harga jual untuk tetap bisa bersaing dengan produk sejenis.
- Membuat produk kustom untuk suatu pelanggan sehingga benar-benar sesuai dengan kebutuhan masing-masing pelanggan, untuk membangun loyalitas pelanggan.

d. Strategi WT

- Selalu mengadopsi teknologi terbaru dengan cepat.
- Selalu melakukan pembaharuan dan memperbaiki kesalahan atau kekurangan produk sebelumnya dan menjadikan hal ini sebagai bentuk jaminan dukungan terhadap pelanggan.
- Menemukan arah atau tren pengembangan baru yang belum dilakukan oleh pesaing.

Dari berbagai kriteria analisa SWOT, bisa diambil kesimpulan bahwa pengembangan software Visualisasi Tiga Dimensi Real Time masih bisa dikategorikan sebagai sesuatu yang layak untuk dikembangkan.

4. IMPLEMENTASI DAN PEMBAHASAN

Software Visualisasi Tiga Dimensi Real Time ini membutuhkan spesifikasi hardware dan software tertentu untuk bisa berjalan optimal. Meskipun OpenGL didukung oleh berbagai platform hardware dan software, dalam implementasi kali ini, dikhususkan untuk penggunaan dalam lingkungan platform x86 32 bit.

4.1. Spesifikasi Hardware

Hardware yang diperlukan untuk bisa menjalankan software/aplikasi visualisasi tiga dimensi real time ini secara optimal sebagai berikut.

- a. Mainboard dengan Chipset Intel, VIA, nVidia, atau lainnya yang mendukung processor generasi 686 keatas.
- b. Processor minimal setara Pentium IV 1.8Ghz, baik Intel, AMD, Cyrix, Transmeta 32 bit atau 64 bit.
- c. Memory minimal 512 MB baik DDR I, DDR II atau DDR III.
- d. Harddisk space sisa minimal 1 GB.
- e. Kartu grafis (graphic card) minimal memory 128 MB dengan dukungan akselerasi hardware terhadap OpenGL (*hardware accelerated*) dengan chipset Intel, nVidia, Gforce, ATI.
- f. Power supply 350 Watt standar.
- g. Kartu Ethernet 10/100 Mbps.
- h. Switch/Hub 10/100Mbps.

4.2. Spesifikasi Software

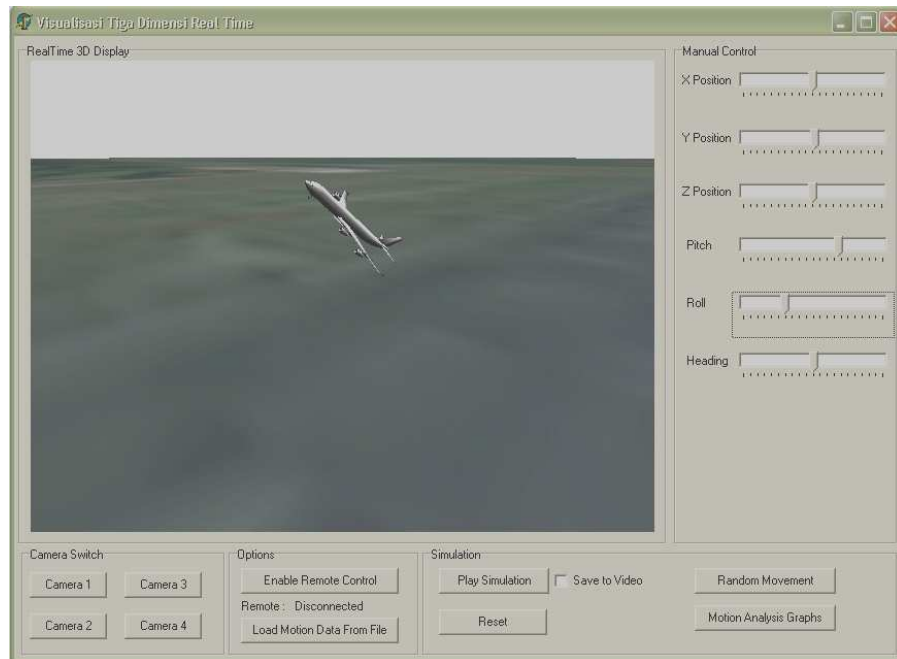
Sistem operasi dan software yang diperlukan untuk bisa menjalankan aplikasi Visualisasi Tiga Dimensi Real Time ini dengan optimal adalah.

- a. Sistem Operasi Microsoft Windows XP 32 bit baik versi Home maupun Professional.
- b. Driver OpenGL minimal versi 1.4, lebih disarankan menggunakan driver OpenGL yang dipaketkan dengan kartu grafis (graphic card) yang digunakan, untuk bisa menggunakan akselerasi OpenGL secara hardware.
- c. Sebagai alternatif bisa digunakan driver MESA 3D sesuai kartu grafis yang digunakan.

Untuk memastikan apakah software dan hardware yang digunakan support OpenGL dan mengetahui fitur-fitur OpenGL yang didukung oleh sistem tersebut (baik hardware maupun software), bisa digunakan software GLInfo yang bisa diperoleh di <http://www.delphi3d.net/hardware>. Mengingat OpenGL memiliki fitur yang cukup banyak, produsen hardware umumnya tidak menyematkan semua fitur tersebut dalam produk mereka, khususnya untuk pasaran menengah ke bawah. Fitur-fitur OpenGL dengan tingkat kompleksitas dan kualitas rendering tinggi umumnya disematkan hanya di produk-produk high end dari kartu grafis mereka tentunya dengan harga yang cukup mahal.

Ketika pengembangan software telah selesai, diperlukan tes terhadap fitur-fitur yang tersedia. Beberapa tes yang dilakukan secara berurutan antar lain.

- a. Tes pertama yang perlu dilakukan adalah dengan menjalankan aplikasi dan memastikan semua elemen yang ada pada mode desain (design mode) terlihat dan sesuai dengan desain. Hasil tes menunjukkan semua terlihat seperti yang ada pada desain antar muka.
- b. Tes kedua adalah melakukan kontrol manual dengan merubah posisi slider TrackBar masing-masing parameter dan juga perpindahan kamera. Semua berfungsi sesuai dengan desain sistem.



Gambar 4.1 Tes Implementasi Sistem

- c. Tes ketiga adalah melakukan input data file CSV berisi data parameter obyek yang disimpan di komputer lokal, kemudian menjalankan simulasi. Hasil tes

menunjukkan parameter dalam file CSV bisa diinterpretasikan secara berurutan atau sekuensial untuk merubah properti obyek tiga dimensi.

- d. Tes keempat adalah dengan membuka jendela grafik analisis. Hasil tes menunjukkan grafik analisis sesuai dengan parameter dari file CSV yang dibaca untuk menggerakkan simulasi.
- e. Tes kelima adalah dengan mengaktifkan tombol Enable Remote Control untuk menyiapkan server bagi aplikasi remote control supaya bisa mengirim data parameter, kemudian menjalankan aplikasi remote control baik pada komputer lokal yang sama maupun komputer lain yang terhubung melalui jaringan TCP/IP, memulai koneksi ke server dan merubah slider TrackBar parameter pada aplikasi remote control. Amati tampilan 3D pada software utama/server, obyek harus berubah secara seketika ketika terjadi perubahan data parameter pada aplikasi remote control. Hasil dari tes ini adalah semua berfungsi dengan baik sesuai dengan desain sistem.

5. KESIMPULAN

Dengan melakukan pengembangan software Visualisasi Tiga Dimensi Real Time Menggunakan OpenGL ini, bisa diambil beberapa kesimpulan, antara lain.

- a. Untuk bisa menampilkan visual tiga dimensi (3D) tidak selamanya diperlukan proses rendering ke file terlebih dahulu, visual tiga dimensi bisa langsung ditampilkan secara real time dengan bantuan library OpenGL, DirectX/Direct3D. Pada umumnya, hampir semua software authoring (modelling/animation) tiga dimensi memerlukan proses rendering yang cukup lama untuk sebuah scene (adegan) yang kompleks.
- b. Dengan bantuan OpenGL atau DirectX/Direct3D bisa membantu mempercepat proses pengambilan keputusan dengan melihat visualisasi secara menyeluruh secara real time dalam format tiga dimensi (3D).
- c. Sebuah visual tiga dimensi (3D) bisa dikontrol secara real time baik secara lokal maupun melalui media serial/parallel atau jaringan TCP/IP.
- d. Sebuah protokol komunikasi sangat diperlukan untuk menjamin transfer data antara client dan server tidak terjadi kesalahan interpretasi.
- e. Menangkap (capture) gambar secara berurutan untuk dijadikan video tidak harus menggunakan hardware khusus, bisa juga dengan cara menangkap memory buffer tampilan dari kartu grafis.

- f. Sistem dunia virtual (virtual world) dalam tiga dimensi (3D) memiliki parameter-parameter sesuai dengan dunia nyata, termasuk tata lampu, sudut pandang kamera, gaya gesek benda, massa benda, gaya tumbuk (collision) antar benda, dan lain sebagainya.

DAFTAR PUSTAKA

- Akenine-Moller; Haines (2002). Real-time Rendering (2nd ed.). AK Peters. ISBN 1-56881-182-9.
- Angel, Edward (2004). OpenGL: A Primer. Pearson/Addison Wesley. ISBN 0321237625, 9780321237620
- Cantu, Marco (2001). Mastering Delphi 6. John Wiley and Sons. ISBN 0782128742, 9780782128741
- Casad, Joe (2004). Sams teach yourself TCP/IP in 24 hours. Sams Publishing. ISBN 0672325659, 9780672325656
- Shreiner, Dave (2004). OpenGL programming guide: the official guide to learning OpenGL, version 1.4. Addison-Wesley. ISBN 0321173481, 9780321173485
- <http://www.opengl.org/>, diakses tanggal 10 Februari 2010
- [http://en.wikipedia.org/wiki/Rendering_\(computer_graphics\)](http://en.wikipedia.org/wiki/Rendering_(computer_graphics)) , diakses tanggal 10 Februari 2010
- <http://glscene.sourceforge.net/wikka/GlsceneHelp>, diakses tanggal 10 Februari 2010
- http://en.wikipedia.org/wiki/SWOT_analysis, diakses tanggal 10 Februari 2010